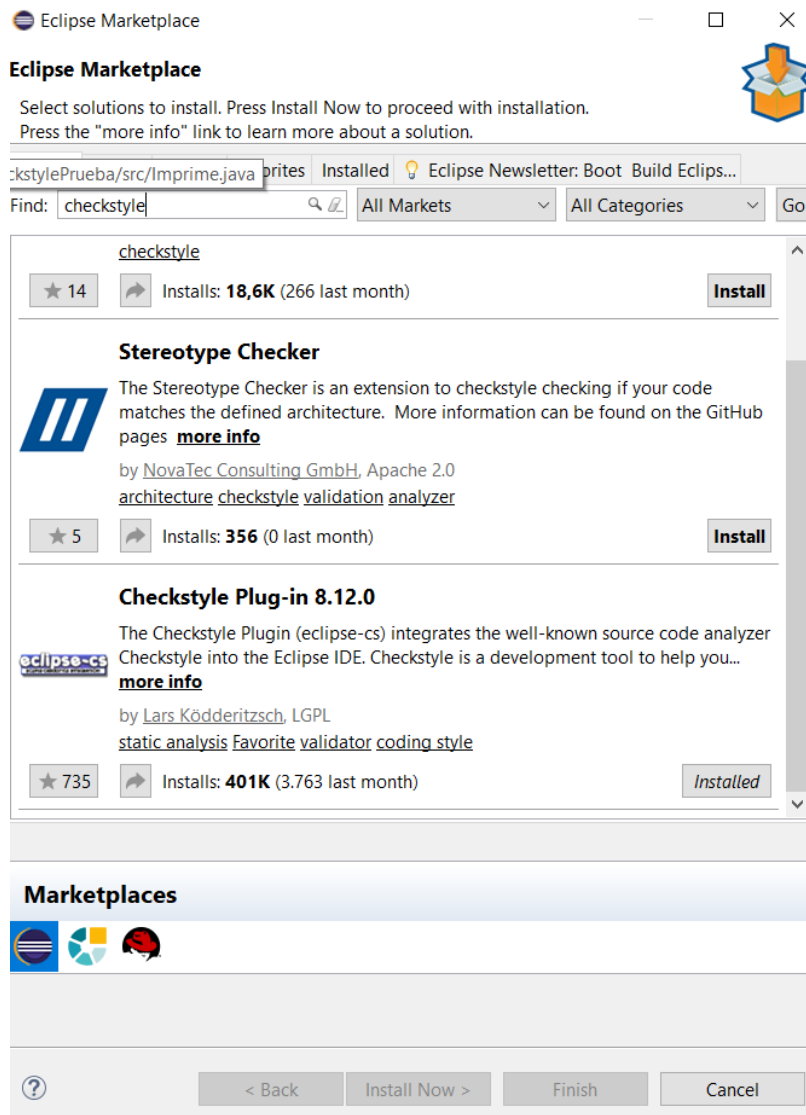


## Tutorial Checkstyle

### CheckStyle en Eclipse

Abrimos eclipse y pulsamos en help-> eclipse- market place.

En la Ventana desplegada buscamos checkstyle y pulsamos en instalar checkstyle-cs.



Descargamos el sencillo proyecto para probar checkstyle del repositorio del taller.

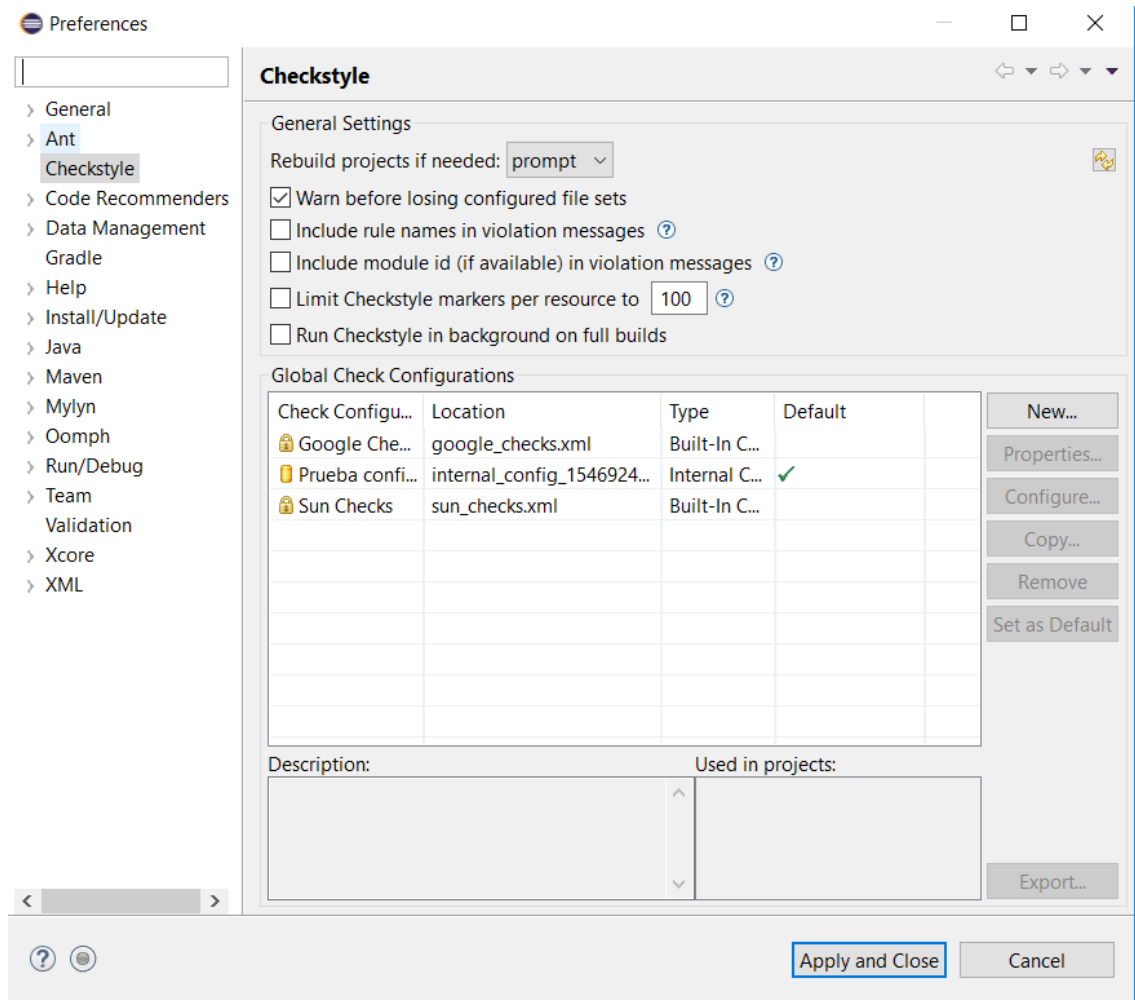
Tras haber abierto el proyecto en eclipse, pulsamos botón derecho sobre el proyecto o paquete que queremos analizar y pulsamos checkstyle -> check code with checkstyle.

Reglas que utiliza por defecto:

<https://google.github.io/styleguide/javaguide.html>

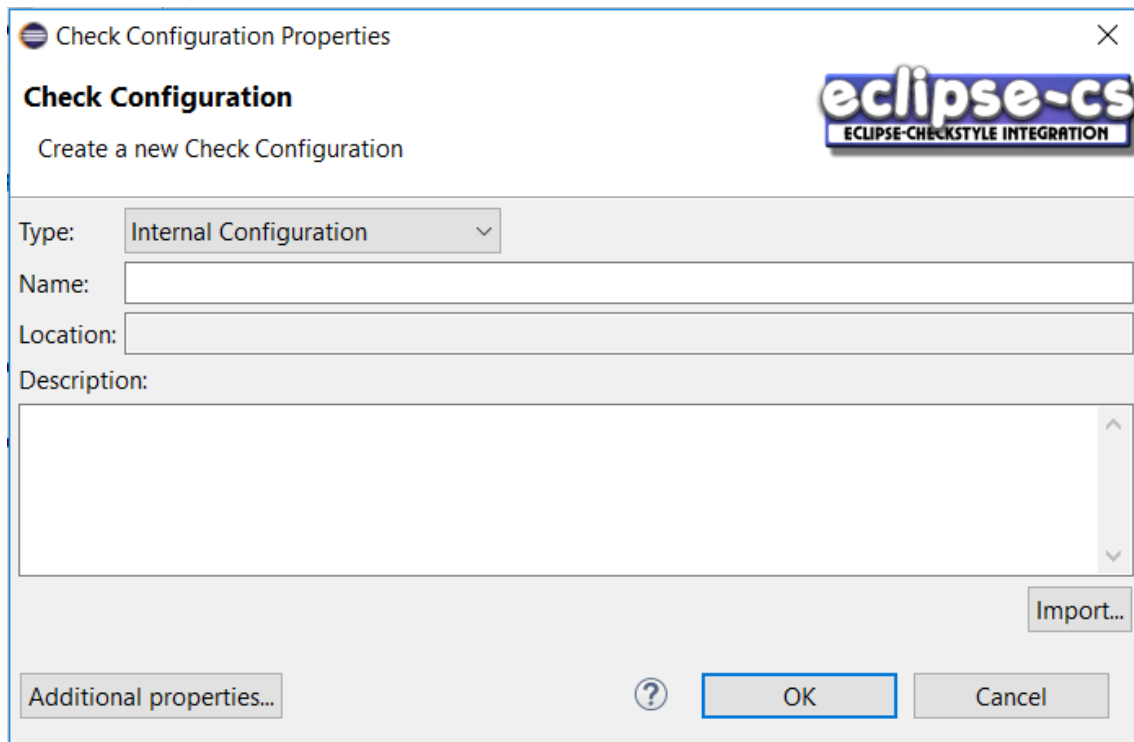
## Configurar nuestras reglas en CheckStyle en Eclipse

Pulsamos window-> preferences->checkstyle



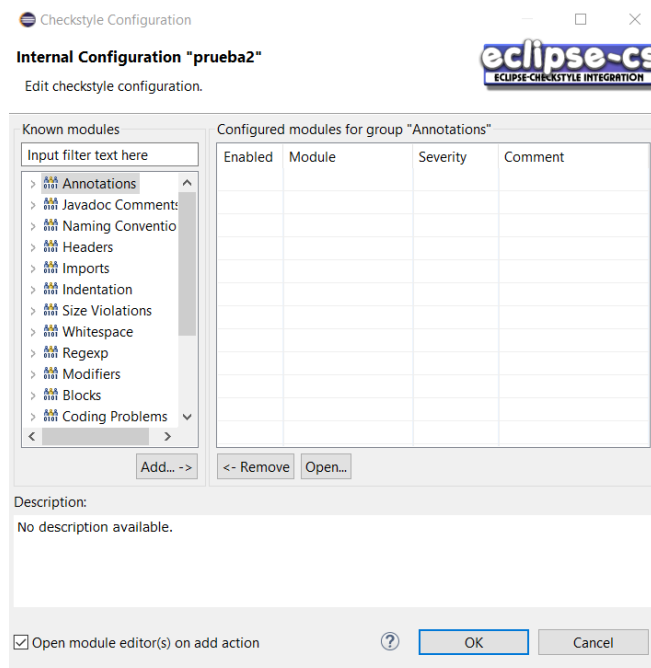
Podemos ver que reglas esta siguiendo actualmente el checkstyle y nosotros podemos añadir nuestra propia configuración haciendo lo siguiente:

Pulsamos new y se desplegará esta pantalla



Completaremos el nombre. En caso de tener un archivo de configuración valdría con poner su nombre y su localización en nuestro ordenador.

Tras haberlo creado lo seleccionaremos y pulsaremos configurar, desplegando así el siguiente menú.



En este menú podemos seleccionar las reglas que queremos que se apliquen en nuestra configuración. Por ejemplo seleccionamos el campo Size Violations y se nos desplegará el conjunto de reglas de ese tipo. Podemos seleccionar Maximun File Length. Para añadirlo

pulsamos en add. Que desplegará un menú para seleccionar el tamaño máximo permitido de nuestro archivo a analizar.

```
<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE module PUBLIC "-//Checkstyle//DTD Check Configuration 1.3//EN" "https://checkstyle.org/dtds/configuration_1_3.dtd"><!--
  This configuration file was written by the eclipse-cs plugin configuration editor
--><!--
  Checkstyle-Configuration: Prueba configuracion
  Description: none
--><module name="Checker">  <property name="severity" value="warning"/>  <module name="FileLength">    <property name="max" value="2"/>  </module></module>
```

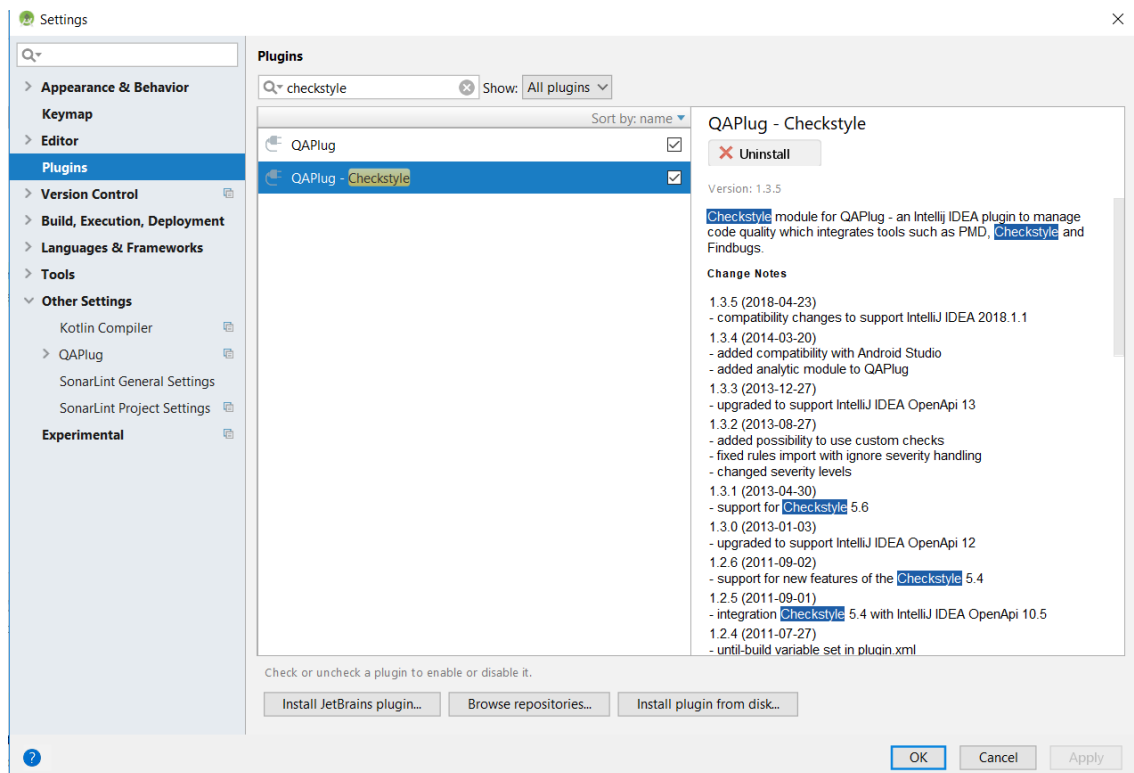
Esto sería la configuración creada anteriormente en la ubicación:

C:\Users\Nombre usuario\eclipse-workspace\.metadata\.plugins\net.sf.eclipsecs.core.

## Introducción a Checkstyle en Android Studio

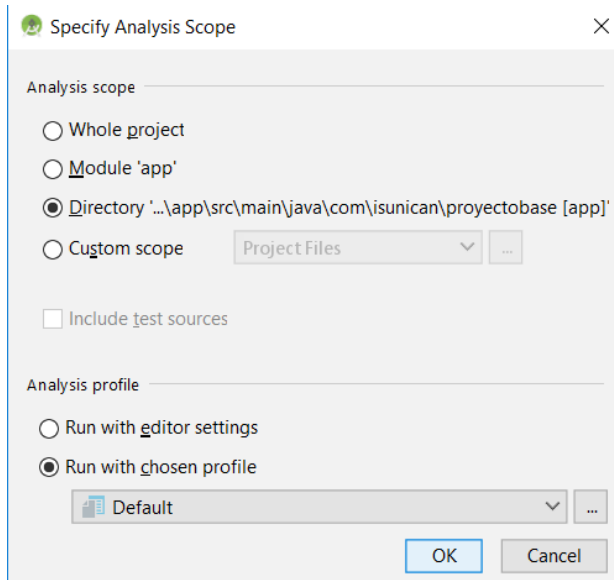
Abrimos el proyecto base-gasolineras utilizado en el repositorio isunican en Android Studio.

Pulsamos en File ->Settings-> plugins y buscamos checkstyle e instalaremos el QAPlug-checkstyle.



Nos instalará también el QAPlug.

Pulsamos botón derecho sobre el proyecto o paquete que queremos analizar y pulsamos en analyze -> analiza code. Esto desplegará esta ventana donde pulsaremos ok.



Veremos los resultados del análisis:

<p>AndroidProject Count: 84</p> <ul style="list-style-type: none"> <li>▼ <b>Maintainability</b> Count: 11 <ul style="list-style-type: none"> <li>&gt; Cyclomatic Complexity Count: 1</li> <li>&gt; Visibility Modifier Count: 10</li> </ul> </li> <li>▼ <b>Reliability</b> Count: 55 <ul style="list-style-type: none"> <li>&gt; Design For Extension Count: 32</li> <li>&gt; Magic Number Count: 23</li> </ul> </li> <li>▼ <b>Usability</b> Count: 18 <ul style="list-style-type: none"> <li>&gt; Final Class Count: 1</li> <li>&gt; Hidden Field Count: 16</li> <li>▼ Method Name Count: 1 <ul style="list-style-type: none"> <li>Gasolinera Count: 1 <p>El nombre 'GetIdeess' debe coincidir con el patrón '[a-z][a-zA-Z0-9]*\$'. <b>Invalid node</b></p> </li> </ul> </li> </ul> </li> </ul>	<p><b>Project:</b> AndroidProject</p> <p><b>Scope:</b> Directory '...\app\src\main\java\com\isunican\proyectobase [app]'</p> <p><b>Profile:</b> Default</p> <p><b>Results:</b> Enabled coding rules: 31</p> <ul style="list-style-type: none"> <li>• Checkstyle: 31</li> </ul> <p><b>Problems found: 84</b></p> <ul style="list-style-type: none"> <li>• Checkstyle: 84</li> </ul>
--	--